What Is Claimed Is:

1. A processor, comprising:

a plurality of pipelined functional units for executing instructions;

a scheduler, coupled to the plurality of functional units, programmed for independently mapping instructions, received from at least two separate instruction groups, to at least a portion of the functional units during a first stage;

wherein the scheduler is programmed to merge and remap the instructions to at least a portion of functional units, based on functional unit requirements and availability, during a second stage.

2. The processor of claim 1, wherein the scheduler is programmed to deliver the instructions to the portion of functional units following merging and remapping.

3. The processor of claim 1, wherein the scheduler, in alignment with a processor clock cycle, is programmed to map the instructions during a first stage of the pipeline for the functional units, and programmed to merge and remap the instructions during a second stage of the pipeline for the functional units.

4. The processor of claim 1, wherein the functional units execute an increased number of instructions operating at a given clock rate.

5. The processor of claim 1, wherein the instruction groups follow a simultaneous multi-threading structure.

6. The processor of claim 1, wherein the instruction groups are prioritized to prevent pipeline failures during execution of instructions.

7. A machine-readable medium having stored thereon a plurality of executable

instructions, the plurality of instructions comprising instructions to:

map instructions, received from at least two separate, independent instruction groups, to at least a portion of a plurality of pipelined functional units during a first stage;

merge and remap the instructions to at least a portion of functional units, based on functional unit requirements and availability, during a second stage.

8. The medium of claim 7, wherein said instructions include instructions to deliver the instructions to the portion of functional units following merging and remapping.

9. The medium of claim 7, wherein said instructions include instructions to map the instructions during a first stage of the pipeline for the functional units, and to merge and remap the instructions during a second stage of the pipeline for the functional units wherein the first and second stage are in alignment with a processor clock cycle.

10. The medium of claim 7, wherein the instructions include instructions to execute an increased number of instructions at a given clock rate.

11. The medium of claim 7, wherein the instruction groups follow a simultaneous multi-threading structure.

12. The medium of claim 7, wherein the instruction groups are prioritized to prevent pipeline failures during execution of instructions.

13. A method for dispersing instructions to executed by a processor, comprising:

mapping instructions, received from at least two separate, independent instruction groups, to at least a portion of a plurality of pipelined functional units during a first stage; and

merging and remapping the instructions to at least a portion of functional units,

based on functional unit requirements and availability, during a second stage.

14. The method of claim 13, further comprising:

delivering the instructions to the portion of functional units following merging and remapping.

15. The method of claim 13, wherein the step of merging and remapping includes merging and remapping the instructions to the portion of functional units to allow execution of an increased number of instructions at a given clock rate.

16. The method of claim 13, wherein the step of mapping includes mapping the instructions, in alignment with a processor clock cycle, during a first stage of the pipeline for the functional units, and merging and remapping the instructions, in alignment with the processor clock cycle, during a second stage of the pipeline for the functional units.

17. The method of claim 13, wherein the instruction groups follow a simultaneous multi-threading structure.

18. The medium of claim 13, wherein the instruction groups are prioritized to prevent pipeline failures during execution of instructions.